RingIt: Ring-Ordering Casual Photos of a Temporal Event

HADAR AVERBUCH-ELOR and DANIEL COHEN-OR
Tel Aviv University

The multitude of cameras constantly present nowadays redefines the meaning of capturing an event and the meaning of sharing this event with others. The images are frequently uploaded to a common platform, and the image navigation challenge naturally arises. We introduce *RingIt*: a spectral technique for recovering the spatial order of a set of still images capturing an event taken by a group of people situated around the event. We assume a nearly instantaneous event, such as an interesting moment in a performance captured by the digital cameras and smartphones of the surrounding crowd. The ordering method extracts the *K-nearest neighbors* (KNN) of each image from a rough all-pairs dissimilarity estimate. The KNN dissimilarities are refined to form a sparse weighted Laplacian, and a spectral analysis then yields a ring angle for each image. The spatial order is recovered by sorting the obtained ring angles. The ordering of the unorganized set of images allows for a sequential display of the captured object. We demonstrate our technique on a number of sets capturing momentary events, where the images were acquired with low-quality consumer cameras by a group of people positioned around the event.

Categories and Subject Descriptors: I.3.5 [**Computer Graphics**]: Computational Geometry and Object Modeling—*Geometric algorithms and systems*

General Terms: Algorithms

Additional Key Words and Phrases: Image alignment and registration, event and action recognition, motion capture and synthesis, image-based modelling.

ACM Reference Format:

Hadar Averbuch-Elor and Daniel Cohen-Or. 2015. RingIt: Ring-ordering casual photos of a temporal event. ACM Trans. Graph. 34, 3. Artilce 33 (April 2015) 11 pages
DOI: http://dx.doi.org/10.1145/2735628

1. INTRODUCTION

The recent smartphone phenomenon changed our lives in numerous ways. In today's world, people are inseparable from their smartphones, bringing them to work, parties, vacations, family gatherings, etc. The purpose of smartphones has extended beyond communication to documenting our lives with visual means. The abundance and accessibility of photos have inspired new applications and catalyzed respective research. Their main theme is to exploit the many different views of the same scene to better understand or reconstruct the scene (e.g., Snavely et al. [2008]).

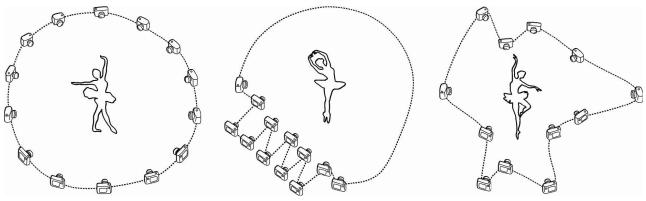

Fig. 1. An illustration of three settings that are applicable for the *RingIt* method.

The wealth of smartphones can also be used to better capture a temporal event. A group of people present during a temporal event can generate a set of photos that can be used for an enriched visual expression. Our work is inspired by the recent work of Basha et al. [2012] who presented a photo sequencing technique. Assuming that a set of still images were captured roughly from the same viewpoint, they determine the temporal order of the uncalibrated images.

In this article, we deal with a different setting: we relax the similar viewpoint assumption and instead assume that a given unorganized set of photographs captures an almost instantaneous event. Rather than temporal ordering, here the endeavor is to spatially order the photos that capture the event. The multitude of cameras redefines the meaning of capturing an event. A dynamic event could be a performance captured by the digital cameras of audience members or, alternatively, the blowing out of birthday candles captured by the smartphones of many relatives and friends. These special moments are often captured by still photographs rather than video footage, in spite of their dynamic nature. As the images are frequently uploaded to some common platform, the image navigation challenge naturally arises. The collaborative photo albums available on Facebook, Picasa, etc., are gaining increasing popularity, and an ordering of the *crowd* photos is a first step to their analysis. We therefore propose *RingIt* for recovering the spatial order of a set of images capturing an event taken by a group of people situated around the event.

Our method accepts a set of rather low-quality images capturing a dynamic event from a variety of viewpoints as input. An event, for our purposes, denotes a momentary occurrence of interest. The relative placements of the digital cameras (or smartphones) capturing the event are unknown, as are their internal parameters. Rather than navigating through the images arbitrarily or according to some geometrical constraints in 2D, we would like to explore the images along a ring centered about the event, where their relative locations along the ring portray the respective points-of-view. The rationale for ordering the images as a ring is that this is the natural layout when people are positioned around an object. Our objective is to recover the spatial order of the images, and hence their ring topology. Figure 1 depicts a variety of different settings that are applicable to our method, along with their respective ring topologies. Notice that by a "ring" we refer to any star-shaped closed curve.

Authors' addresses: H. Averbuch-Elor (corresponding author) and D. Cohen-Or, Tel Aviv University, Tel Aviv-Yafo, Israel; email: hadar.a.elor@gmail.com.

DOI: http://dx.doi.org/10.1145/2735628

(a) (b) (c) (d)

Fig. 2. Overview of our method. (a) Given a set of unorganized casual images capturing a dynamic event, we first (b) compute a full all-pairs dissimilarity matrix. Next, (c) we define a sparse refined dissimilarity among the KNN. Finally, (d) a spectral analysis reveals the ring order of the images.

Structure from motion (SfM) techniques compute the camera parameters and the 3D structure for a set of images simultaneously and could ultimately recover the ring topology. However, the success of the technique requires the existence of a dense set of photographs that can support accurate correspondences to compensate for the vast number of unknowns, including those of the reconstructed model. In our work, we are not interested in reconstructing the scene nor the camera poses, but rather in recovering the spatial order. We demonstrate successful ordering of relatively sparse and low-quality image sets that cannot be recovered using standard SfM techniques.

Our method orders the images by using a weighted Laplacian [Belkin and Niyogi 2003]. Graph Laplacians are widely used in machine learning for various applications, most commonly for dimensionality reduction or spectral clustering. To our best knowledge, they have not been used for an ordering problem, in particular for ordering crowd photos. Our approach to spatially sort the image set operates in two stages: (i) constructing a sparse dissimilarity matrix, followed by (ii) a spectral analysis and ring ordering, as illustrated in Figure 2. The all-pairs image dissimilarities are first computed quickly, and then the KNN of each image is extracted. Since large distances are unlikely to be reliable, they are removed and the KNN are refined by a more accurate dissimilarity measure. In stage (ii), a spectral analysis of the sparse dissimilarity matrix yields a *ring angle* for each image. The spatial order is then recovered by sorting the obtained ring angles.

Numerous methods that aim at extracting information from shared image content could potentially benefit from RingIt. For example, techniques that propagate editing among photo collections (e.g., HaCohen et al. [2013] and Yücer et al. [2012]) immediately gain from having them ordered. Certainly, a spatial ordering is also helpful as a means of visualization. The ordering can further allow for an interactive "bullet-time" display of an unstructured set which departs from the scrupulous viewpoint trajectory of Hollywood productions or the lightweight video-based system [Zitnick et al. 2004].

The rest of the article is organized as follows. In Section 2 we discuss a number of related works which are most relevant to this article. In Section 3 we present the general concept behind our dissimilarity measure and specifically consider two types of distances, image-based and contour-based, showing that our method is not constrained to specific distances (the implementation details are discussed only later in Section 5). The ordering method is presented in Section 4. For completeness, general concepts in spectral graph theory are discussed as well. In Section 6, we present a completely unsupervised technique that creates a bullet-time display of the unstructured image set based on our ring ordering. Section 7 presents extensive experimental results. Lastly, we conclude with a discussion of limitations and future work in Section 8.

2. RELATED WORKS

Photo organization and navigation. Organizing a set of images captured in an uncontrolled environment has received large interest recently. Driven by the ever-growing accessibility to immense image repositories and the increasing popularity of geo-tagged photos, new avenues have opened with countless possibilities (e.g., Lu et al. [2010]). Kemelmacher-Shlizerman et al. [2011] organize unstructured collections of the same person using a shortest path on a graph. Our work is more closely related to those that analyze smaller collections of images capturing the same scene. Wan et al. [2012], for example, present a technique of sorting an image set portraying an urban scene, consisting of piecewise-planar geometric primitives. Using single-view geometry and segmentation techniques, they cluster the images and disambiguate symmetries. They also demonstrate that the rough sorting then simplifies the SfM reconstruction.

Spectral approaches are naturally associated with the endeavor to better understand such collections. For example, Heath et al. [2010] investigate the connectivity of a large image collection. By exploiting such spectral tools, they also demonstrate different ways to explore these collections. There are several spectral techniques that explicitly consider cyclic manifolds. Pless and Simon [2001] extend the MDS embedding algorithm to spherical manifolds. Lee and Verleysen [2005] present a technique that tears the manifolds in order to embed the data onto low-dimensional data. Using persistent topology, de Silva et al. [2011] present a procedure for constructing circular coordinates. Our method orders the images by using a weighted Laplacian [Belkin and Niyogi 2003]. Our casual setting departs from the cyclic structures presented in previous works. Unlike a synthetic environment or lab conditions, a collection of casual photographs is more challenging to analyze.

The navigation task is an immediate sequel to the organization problem. Interactive navigation through a large photo collection is analyzed extensively in Snavely et al. [2008]. They also discuss the image-based rendering aspect that allows compelling paths to be created. Kushal et al. [2012] introduce the foundation to Photo Tours in Google Maps, a feature that automatically creates movies out of still photographs through world-famous landmarks. The closely related field of video-based rendering allows for image navigation in 3D. Ballan et al. [2010] demonstrate impressive results of image navigation in dynamic events captured by multiple video cameras. Recently, Arpa et al. [2013] introduced a near-real-time system of photos that are captured casually and then uploaded to a central

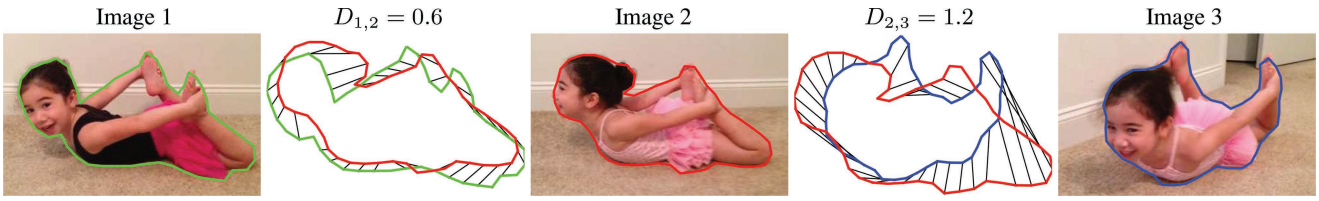

Fig. 3. RingIt from contours. As our ordering method requires only distances, displayed above are those distances obtained from contours only, that is, the images themselves are not used by RingIt. The three images belong to our Backbend set.

server. Unlike our work, they are not interested in ordering all the images spatially, but rather in finding a smooth path in a user-specified direction. By enabling users to mark the object of interest, the images along the path are aligned and a plausible bullet-time effect is achieved.

3D reconstruction of a scene. A key challenge is to reconstruct the geometry latent in the unorganized set of photographs. Recent advances in feature matching techniques led to significant achievements in the field of structure from motion, and these techniques have advanced from reconstructing static scenes captured by a single casual photographer [Schaffalitzky and Zisserman 2002] to reconstructing static scenes from large Internet photo collections [Snavely et al. 2006]. To accelerate the reconstruction of millions of Internet images depicting, for example, the city of Rome, Frahm et al. [2010] first organize the images using a similarity metric that is based on GIST features [Oliva and Torralba 2006]. These features allow for a large-scale clustering, distinguishing between the different scenes. Our work, on the other hand, accepts images all capturing the same scene, and therefore we care for a more precise distance measure.

A 3D reconstruction can also benefit the analysis of dynamic scenes. Since the introduction of this concept by Kanade et al. [1997], prospective research for dynamic 3D reconstruction has flourished, mainly for video broadcasting purposes at sport events. These methods commonly accept a monocular video as input. A recent technique proposed by Guillemaut et al. [2009] performs 3D reconstruction in rugby and soccer games from casual multiple-view videos. There are several other works that aim at deciphering latent information from casually captured videos. Arev et al. [2014], for instance, present a technique that produces a coherent video "cut" from multiple casual videos. In our work we also address such dynamic scenes but, unlike the aforementioned work, only short or instantaneous ones. Furthermore, we assume the scene is captured solely by still photographs.

3. SPARSE DISSIMILARITY MATRIX

The first stage of our method is estimating the dissimilarity between pairs of unordered input images. Measuring dissimilarity between a pair of similar objects is closely related to the problem of matching these objects. A relatively high number of matching features, for example, usually indicates a pair of similar images and therefore should correlate to a lower-dissimilarity measure. The following dissimilarity measure, also known as Dice's coefficient, is based on this concept:

$$D_{i,j} = 1 - \frac{2 \cdot N(I_i, I_j)}{m_i + m_j}, \tag{1}$$

where $N(I_i, I_j)$ denotes the number of matching features, and m_i and m_j denote the number of extracted features in images i and j, respectively. It is important to note that the popular image features are roughly scale and rotation invariant, but by no means viewpoint invariant. Therefore, this type of measure relates directly to our objective of ring ordering. It should also be noted that, if the images share enough content, it might be reasonable to examine the quality of the correspondence or to look for more precise matching techniques. Clearly, more exact measures require more time.

Our method strives to balance between speed and quality. We certainly care for a fine-grain distance that can estimate dissimilarities among those images that capture the same event from just slightly different viewpoints. Nonetheless, a more naive approach can well-enough distinguish which images have almost nothing in common, or at least less in common compared to the others. Thus, an initial dissimilarity measure is determined according to Eq. (1), where $N(I_i, I_j)$ is computed using sparse image descriptors. Later, the large distances are removed and only the smaller ones refined. The refined distances are estimated using dense descriptors. Such dense descriptors are usually more time consuming, yet allow for a more elaborate measure. Often, these descriptors incorporate a confidence measurement for each pixel-to-pixel correspondence. We use these confidence measures to refine the distances. See Section 5 for more details.

To emphasize that our technique does not necessarily rely on image-based features, we also examined our technique on distances that are contour based. In this scenario, the distances are obtained from contours only, that is, without using the images themselves. Similarly to the image-based approach, the initial distances are estimated according to Eq. (1), but using a contour-matching technique rather than a feature-matching one. These distances are refined by examining the L1-sum of the normalized matching 2D points. We normalize the matching 2D points as follows: we first translate each contour so its center of gravity is at the origin and then scale it so that the distance of the contour points to the origin has a unit variance. Figure 3 illustrates our method in this setting. Notice in the figure that the correspondence is performed independently of the pixel values expressing the girl's dress. This is desirable when we aim at ordering a specific stance that does not necessarily occur as an instantaneous event. We evaluated this approach on image sets with plain or repetitive backgrounds, where the contour extraction using interactive segmentation techniques is relatively simple.

4. SPECTRAL ANALYSIS AND RING ORDERING

From the image dissimilarity, we extract the ring ordering. We assume we are given N images capturing an object of interest. Let A be the region of interest capturing this object in the images. Normally, A appears roughly in the center of the frames, but not necessarily. We would like to recover the spatial order of the set of such N unorganized images. Neither their absolute nor relative positions are known, and the cameras are uncalibrated. The objective is to recover the order of the images, and hence their ring topology, from the given image dissimilarities.

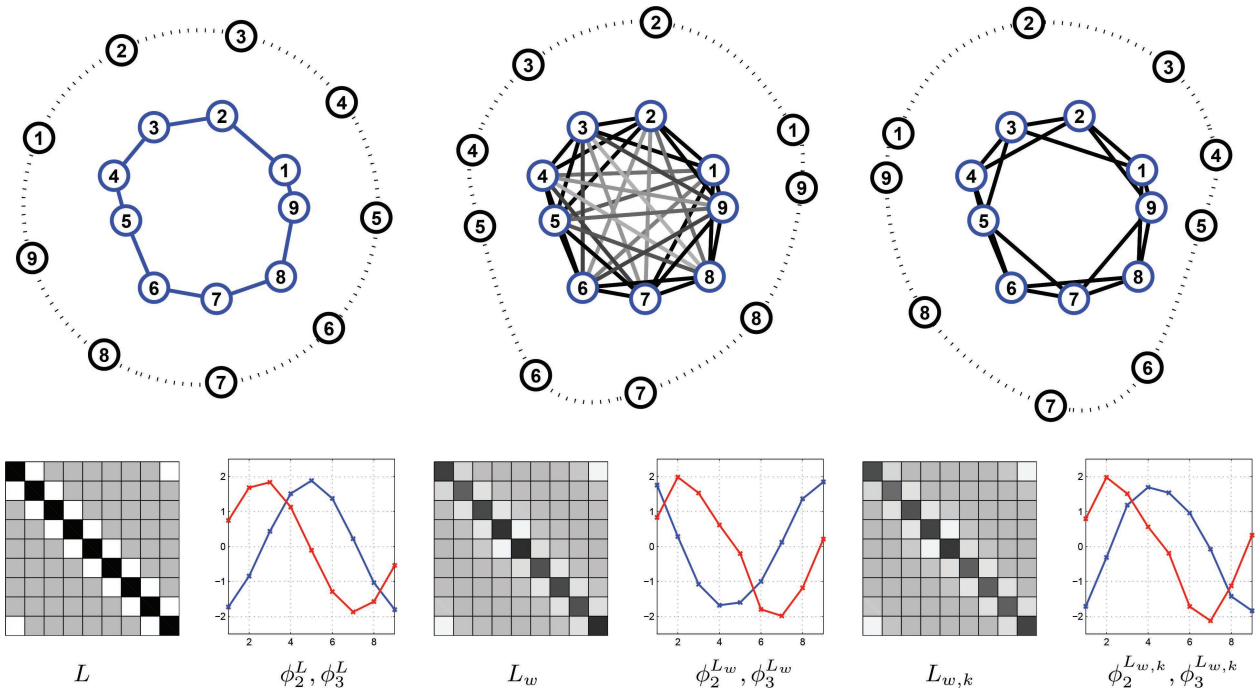

Fig. 4. The spectral approach for ordering N points in $\mathcal{R}^2$. Spectral embedding of $N = 9$ input points (blue) lying approximately along a ring. The resulting embeddings are presented by the exterior rings. For the matrix illustrations, the color values range from -1 (white) to 2 (black). Left: Assuming adjacent neighbors are known (unweighted edges colored in blue) yields the geometrically oblivious L, whose eigenvectors are a sine and cosine. Center: The Euclidean distances are considered for the weighted graph modelling (darker edges correspond to smaller distances). The embedded points, or the 2^{nd} and 3^{rd} eigenvectors of L_w, reside roughly along a ring. Right: Only the KNN ($K = 4$) of each point is considered for the construction of the sparse matrix $L_{w,K}$, whose eigenvectors resemble those of L.

For simplicity, in this section we shall assume our input consists of N two-dimensional points. The extension to the high-dimensional images is immediate, as it only affects the inter-point distance computation. The problem of finding an optimal path along all given points from their inter-point distances, or dissimilarities, has been studied extensively, and commonly nests under the title of the *traveling salesman problem*, or TSP for short. Assuming the ring order naturally defines the shortest path, TSP becomes extremely relevant. Yet, TSP is NP-hard and furthermore hard to approximate when the triangle inequality is not guaranteed, as in our setting. To enforce the triangle inequality one can apply a classical MDS and embed the points in an Euclidean space, and then apply TSP on those points embedded in the enhanced space. Nevertheless, as we shall show in the following, under the assumption that the points were sampled roughly along a ring, their order can be immediately recovered by a spectral analysis over a sparse dissimilarity matrix where each row includes only the K-nearest neighbors.

First, let us show how a spectral analysis can be used to solve a much simpler ordering problem, even a trivial one. Assume that each point knows a priori its two adjacent neighbors along the ring. Recovering the order along the entire ring is a trivial task as the partial relations overlap and one can simply reconstruct the entire sequence by stepping from one point to its known adjacent point, forming the entire sequence along the ring. Nevertheless, let us examine how the order can be recovered using a spectral analysis. Then, the strong requirement of knowing the immediate neighbors will be relaxed.

Spectral Analysis. Let $G(V, E)$ be an undirected graph where each edge $e \in E$ connects two vertices which are adjacent on the ring. Let A_G be the adjacency (connectivity) matrix of the graph

$$A_G(i, j) = \begin{cases} 1 & \text{if } (i, j) \in E \\ 0 & \text{otherwise,} \end{cases}$$

and let D_G be the degree matrix of the graph:

$$D_G(i, j) = \begin{cases} 2 & \text{if } i = j \\ 0 & \text{otherwise.} \end{cases}$$

Generally speaking, the (combinatorial) Laplacian matrix of a graph is a simple function of its adjacency matrix A and its degree matrix D: $L = D - A$, and with $A = A_G$ and $D = D_G$, the eigenvectors of L belong to the well-known DCT basis [Brouwer and Haemers 2012]. The first eigenvector of L is a constant and the second and third are a sine and cosine. The second and third eigenvectors provide an embedding of the input points, which are not necessarily on a circle, into a circular ring, while maintaining their spatial order. More precisely, let ϕ_2 and ϕ_3 be the 2^{nd} and 3^{rd} eigenvectors, and let ϕ_2^i and ϕ_3^i be the i^{th} pair, that is, the coordinates of the i^{th} embedded point, then $(\phi_2^i, \phi_3^i) = (cos\,(2\pi i/N)\,, sin\,(2\pi i/N))$, with N denoting the number of input points. A sorted L and its 2^{nd} and 3^{rd} eigenvectors are displayed in Figure 4. By "sorted", we mean that adjacent vertices are also consequentially enumerated, as illustrated in the top row of Figure 4. A permuted L would yield

similar behaviour, but would be less immediately apparent. Note that L accepts three values only:-1(white), 0(gray), and 2(black).

In our problem setting, we do not seek for an embedding of the points but merely their order along the ring. Therefore, we would like to extract the angle of the embedded points about the origin, namely the ring angles. A simple function of the 2^{nd} and 3^{rd} eigenvectors of L will yield the desired ring angles. More formally, the ring angle $\theta_i \in [-\pi, \pi]$ is defined according to $\theta_i = \text{arctan2}(\phi_3^i, \phi_2^i)$, where

$$\text{arctan2}(y, x) = \begin{cases} \arctan \frac{y}{x} & x > 0 \\ \arctan \frac{y}{x} + \pi & y \geq 0, x < 0 \\ \arctan \frac{y}{x} - \pi & y < 0, x < 0. \end{cases}$$

The spatial order can then be recovered by sorting the obtained ring angles.

The Relaxed Problem and L_w. Now, consider the edge-weighted graph, that is, the pair (G, w) where $w : E \to \mathcal{R}$ is a weight function. Let A_w be the adjacency matrix of the weighted graph:

$$A_w = \begin{cases} w(i, j) & \text{if } (i, j) \in E \\ 0 & \text{otherwise.} \end{cases}$$

Following Belkin and Niyogi [2003], the edge weights are given by

$$w(i, j) = \exp\left\{-\frac{\|d_{i,j}\|^2}{t}\right\},$$

where $d_{i,j}$ is the distance between vertices i and j, and $t \in \mathcal{R}$ is a prescribed parameter. Note that the distances are not necessarily symmetric by nature, but for our analysis we shall assume symmetric relations, that is, $d_{i,j} = d_{j,i}$ for all $i, j \in V$. The weighted graph (G, w) can be seen as an approximation of the unweighted graph G. Although explicit knowledge regarding the immediate neighbors is no longer assumed, the immediate neighbors' small distances yield high weight edges. Note, however, that the weighted graph also contains a large number of edges which do not exist in the unweighted graph. Nevertheless, these edges are more likely to have lower weights relative to those edges that exist in G as well. Thus, in the weighted scenario, adjacent vertices on the ring are implicitly expressed by high weight edges and not explicitly as in the unweighted case.

Let D_w be the diagonal degree matrix of the weighted graph, that is,

$$d_{w_{ii}} = \sum_{i:(i,j)\in E} w(i, j),$$

where $d_{w_{ii}}$ denotes the vertex weight. Let L_w be the weighted Laplacian matrix of the weighted graph, defined by

$$L_w = D_w - A_w.$$

Hall [1970] showed a relation between (ϕ_2, ϕ_3), the eigenvectors of the weighted Laplacian, and the minimization of the following optimization problem:

$$argmin_{x,y} \sum_{(i,j)\in E} w(i, j) \cdot \| (x(i), y(i)) - (x(j), y(j)) \|^2.$$

This minimization indicates the eigenvectors of L_w provide the best 2D embedding in the sense that the original distances are preserved as much as possible, that is, smaller distances in the embedding correspond to "heavier" edges.

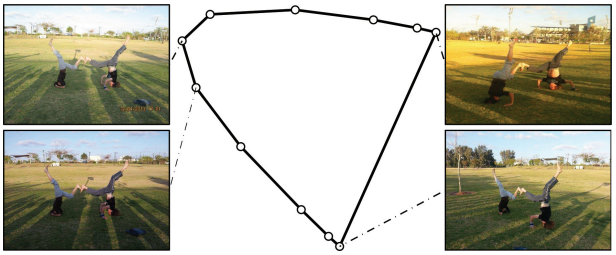

Fig. 5. Example of an embedding. The embedding of the HEADSTAND set, captured along a semi-circle. The two images on the right are from very different viewpoints, while the two on the left are quite similar. This is clearly reflected in their embedding.

Since it is assumed that the points are placed roughly along a ring, the resulting second and third eigenvectors are close to the sine and cosine functions, as in the unweighted case. In other words, an embedding using the spectral basis of L_W yields a good approximation to an embedding to a perfect circle. Hence the order of the points can be calculated by extracting the 2^{nd} and 3^{rd} eigenvectors and sorting the ring angles. As long as the sampled points are close to a circular ring, so are the embedded points, and the ring angles yield a plausible sorting. However, as the points depart from a circle and thus from our setting, the quality of the ordering diminishes.

Now, rather than using the full matrix L_w, we can use a sparse matrix L_w^K where the edge weight $w_{i,j}$ is considered only if j is among the KNN of i, or if i is among the KNN of j. This bidirectional neighborhood relation forces L_w^K to be symmetric and positive definite, thus maintaining one of the most basic properties of the Laplacian. Notice that if L_w^K is an approximation of L_w, then its spectral eigenvectors approximate the ones of L_w, and ultimately approximate the ones of L which are a sine and a cosine. In other words, since the KNN includes the K-nearest neighbors of each point, the points are likely connected to their two adjacent points along the latent ring connectivity that we are after. This approximation is illustrated in Figure 4. The Euclidean pairwise distances can be considered for the weighted graph modelling. By considering only the KNN ($K = 4$), a sparse $L_{w,K}$ is constructed and yields an embedding on a geometry that is close to a circular ring, since its eigenvectors are an approximation to the well-known DCT basis.

The distances between graph vertices, in our problem setting, are the dissimilarities between the images. The KNN of an image could then be determined by a quick and rough dissimilarity estimate, and thus spare a more accurate, yet time-consuming, dissimilarity measure for the image with its close neighbors only. Figure 5 illustrates an embedding of 11 input images. The figure illustrates that an analysis of the sparse matrix yields a plausible embedding that reflects the dissimilarities among all images.

5. IMPLEMENTATION DETAILS

Two sparse descriptors, SIFT and SURF, were evaluated for providing initial distances among the image pairs. As we are only interested in a rough dissimilarity measure and without insisting on recovering the exact temporal accuracy, the sparse feature matching is quite naive. A match is determined by applying a threshold to the ratio between the distance of the closest neighbor to that of the second-closest neighbor. If the ratio exceeds a prescribed threshold, the features are considered unique and the match is accounted for. As shown in the Supplementary Material accessible in

Fig. 6. Two examples of the spatial ordering obtained by RingIt. Presented above are the ring orderings of six consecutive images from the sparse THIGH-STAND set (top) and 12 images sampled from the ARM BALANCE set (bottom). The ring ordering is demonstrated clockwise. The other photographers shooting the event simultaneously can be observed in the background. Note that some of the photographers captured the event rather off-center.

the ACM Digital Library, SURF is generally more successful than SIFT. Therefore, we use it as our default basic descriptor.

The *nonrigid dense correspondence* (NRDC) [HaCohen et al. 2011] technique was used in order to retrieve the dense correspondence. NRDC outputs a confidence measurement for each pixel, so Eq. (1) was adapted simply by incorporating a summation of all confidence values into $N(I_i, I_j)$. As symmetry is essential for the spectral analysis and since the output provided by NRDC is not, both $N(I_i, I_j)$ and $N(I_j, I_i)$ were examined and the distance was determined according to the higher confidence measure. The refined dissimilarity matrix is obtained by this confidence measure solely, that is, the distances provided by the sparse descriptors are disregarded. It is important to note that, in case NRDC fails to find any correspondences among the KNN images, the dissimilarity matrix is not refined and the rough estimates are used instead. For example, Thigh-Stand (see Figure 6) demonstrates a successful ring order obtained with SURF alone. The contours were extracted using Grabcut [Rother et al. 2004] and iCoseg [Batra et al. 2010] and matched using the *inner-distance shape context* (IDSC) [Ling and Jacobs 2007] followed by an L1 matching (see Figure 3).

It is important to note that other reasonable distances can be applied successfully. As long as the distance among two images conveys their distance along the ring, it can be easily adapted into our ordering technique. Note, however, that GIST descriptors, for example, are unsuitable for computing the pairwise distances as a large difference in scale can make a big impact on the distance value regardless of the ring distance.

Regarding the number of neighbors used, namely the K, we used $K = 4$ for all our sets. However, to ensure a KNN connectivity, we verified that the second-smallest eigenvalue is greater than zero. Normally, this occurred naturally. Nevertheless, this extra step confirmed that the graph is simply connected, which is necessary in order to evaluate all the images together.

6. UNSTRUCTURED BULLET-TIME DISPLAY

Ordering the unorganized images collection consequently provides means for an unstructured bullet-time display of the event. A bullet time, sometimes referred to as frozen time [Zitnick et al. 2004], is a sequencing of coherent views around an action, where the action seems frozen during the display. To achieve this visual effect, typically an array of calibrated cameras, perfectly synchronized, are set around the event to form a coherent sequence. We refer to our bullet-time display as unstructured, as it is captured by uncorrelated, uncalibrated cameras, of typically low quality. Our ambition is not to create a high-quality bullet-time sequence, but merely to generate a casual order display of the unorganized set of photos along a ring.

The challenge in creating an unstructured bullet-time display is in aligning the photos plausibly. The main challenge in the alignment is that the object is not separated from the background. Clearly, if such a separation exists, then a visualization can be obtained by observing the pixel-to-pixel correspondences along the object and aligning consecutive photos using these correspondences. Arpa et al. [2013] obtained a similar unstructured display using a manually marked bounding box of the object in each of the set images. In the following, we present an alignment technique that is completely unsupervised.

The input to our bullet-time technique is the ring ordering of the image, and the pixel-to-pixel correspondences of adjacent images along the ring. These correspondences were previously computed as described in Section 5. The key idea is that, once we discover correspondences that connect between the object in two adjacent images, then we can propagate these correspondences along the ring. As we do not know which pixel correspondences are associated with the common object of interest, we start at a candidate seed image and attempt to propagate the estimated foreground mask from the seed to the rest of the set.

For each image and its two adjacent neighbors along the ring, we can guess which corresponding pixels belong to the object using the following simple procedure: We assume that the object is somewhat centered and consider a 50×50-pixel window around the center. For each neighboring image, we search for the best similarity transformation of the corresponding pixels within this window. Then, we can expand the initial window and include more pixels that agree (inliers) with this transformation. All the inlier pixels together yield our seed mask of the object.

To propagate the seed mask to the rest of the set, we examine the neighboring images. We can estimate the location of the object in the neighboring images by examining where the candidate mask transfers, using the given pixel-to-pixel correspondences. We then

Table I. Dataset Evaluation

Dataset			RingIt	MDS		SfM	
Name	#img	#ppl		1D	2D	1	2
HEADSTAND	11	11	0	10	0	6*	7*
THIGH-STAND	11	11	0	16	0	0*	2*
BOTTOM-UP	14	3	5	23	21	9*	11*
BIKE&STATUE	21	1	0	54	53	4*	3*
FRONT HANG	21	21	1	43	35	2*	2*
SIT-ON-FEET	21	21	4	65	8	6*	0
CANDLESTICK	21	21	5	81	2	20*	11*
STAND-ON-FEET	21	21	1	65	0	7*	4*
BACK BALANCE	31	21	3	109	6	3*	7*
ARM BALANCE	31	21	5	176	10	4*	3*
FRONT SWAN	35	21	0	177	10	4*	2*
MOTORCYCLE	71	1	0	622	755	21*	0
ARABESQUE	10	10	5	15	5	1*	2*
BALLERINA	15	1	6	33	7	0*	3*
BACKBEND	15	1	8	11	30	1*	2*
FLAMINGO	15	3	5	17	17	1*	2*
BRIDGE	17	2	1	22	8	1*	9*
DEVANT	20	10	5	22	10	4*	4*

The number of images (#img) and the number of participating photographers (#ppl) are portrayed along with the set names. The entries display the swapping distance to the ground truth. The last six sets were examined with the contour-based approach rather than the image-based one. RingIt is compared to MDS&TSP in both 1D and 2D and to two popular SfM open-source codes: Bundler(1) and VisualSfM(2). A star(*) implies that the method failed to incorporate all the images in the set reconstruction, and therefore the swapping distance could not be computed. In such cases, the number of camera locations recovered is displayed alongside the star.

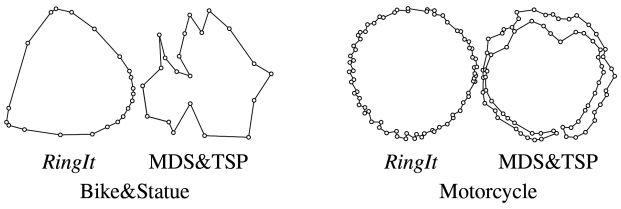

Fig. 7. RingIt vs. MDS&TSP. Above are the embeddings of the sets BIKE&STATUE and MOTORCYCLE obtained by RingIt and MDS&TSP. The ring ordering produced by the methods is presented by the lines connecting the dots. RingIt yields perfect ordering for both sets while MDS&TSP yields 53 and 755 swaps for the BIKE&STATUE and MOTORCYCLE sets, respectively.

look for the best transformation of the estimated object between these images and their neighbors along the ring. A good seed propagates the object mask along the entire ring. To obtain a good seed image quickly, we sort the set images according to how well the centered window agrees with the best transformations to its immediate neighbors. More precisely, we count the number of inliers within this window and weigh these inliers according to their distance from the center:

$$\exp \frac{-d_{inliers}^2}{t^2},$$

where t is a parameter we set to 25 and $d_{inliers}$ is measured in pixel units. On all our sets with precomputed correspondences, a good seed mask was found and the propagation technique allowed for a plausible bullet-time display. Please refer to the attached video clips in the ACM Digital Library for assessing the high quality of our results.

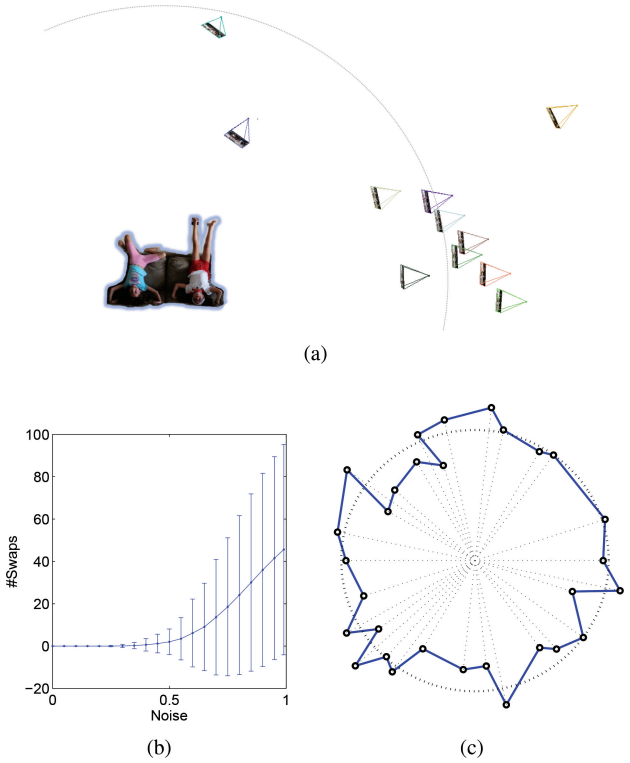

Fig. 8. Sensitivity to noncircular camera placements. (a) An example of a noisy noncircular setting that the method overcomes successfully. (b) The graph demonstrates how well the algorithm performs (in terms of the number of swaps) under noncircular camera locations. (c) A visualization of one of the 10000 synthetic test cases at a noise level $r_t = 0.25$.

7. RESULTS

To evaluate the method, we generated 18 datasets of outdoor and indoor scenes. Almost all of them captured dynamic scenes. A few were captured in plain surroundings in order to examine the contour-based approach. The events were asynchronously captured with a multitude of capture devices including DSLR cameras and mobile phones. Therefore, the input images have significant differences in camera settings and image quality, and temporal differences in acquisition. Furthermore, the input images were typically of low quality, for instance, the resolution of those transferred by means of the WhatsApp application was normally 745-by-556-pixel resolution only. In order to evaluate denser sets, some of the photographers captured nearly identical images. We also experimented with just partial rings. The scenes CANDLESTICK and SIT-ON-FEET were captured from cameras of people standing in two straight lines, one higher than the other (this type of setting is also illustrated by the central drawing in Figure 1). The assumption regarding the event being situated in the center of the frame was also relaxed, as some of our photographers captured the event rather off-center. The lighting condition is poor as well in many of our scenes.

For each set, denote the ground-truth ring order by $I_1, \ldots, I_N$ and the estimated ring order by $I_{i_1}, \ldots, I_{i_N}$, where N is the number of images in the set. The problem of finding the correct ring order is then equivalent to finding the correct image permutation along the ring. Among the $\frac{1}{2}(N-1)!$ possible permutations, only one is correct. To measure the distance from a given permutation to

Fig. 9. RingIt in the wild. Above are ring orderings produced by RingIt of two image sets downloaded from the Internet, demonstrated clockwise. In the set portraying Obama's inauguration (a sample of the set is illustrated above), image-based distances were used. Images courtesy of The White House. The set depicting a Tai Chi kick does not capture an instantaneous event, hence contour-based distances are used. Notice the ordering of the Tai Chi kick is approximately accurate, except for a few apparent swaps on the right.

the correct one, we use the swapping metric [Cormode 2003]. The swapping distance is defined to be the minimum number of swaps, or transpositions, of two adjacent elements necessary to transform one permutation into another.

Table I demonstrates the number of swaps obtained on each of our generated sets. The majority of the sets (12 out of 18) were evaluated by our image-based approach, and the rest were examined by the contour-based approach to show the method is not limited to a specific distance metric. As a sanity check, we also examined the image-based approach on the first two Oxford multi-view datasets [Visual Geometry Group 2004], Dinosaur and Model House. Both sets capture a simple static scene and yield zero swaps using our technique.

Comparison to MDS&TSP. We compared our results with the ring order obtained by first applying a classical MDS on the full dissimilarity matrix, followed by a TSP approximation on the embedded points. We examined embeddings in both 1D and 2D. In most cases, the ordering of the 2D embedding proved more successful, which suggests that a 1D embedding does not provide sufficient information for a spatial ordering. As our empirical experiments demonstrate, the MDS together with TSP optimizes a general path rather than a circular ring. For example, in the MOTORCYCLE set, the input images capture a motorcycle from slightly different angles all around. The optimal path obtained by TSP is to travel across most of the ring along more similar angles and then to travel back along the remaining angles (see Figure 7), yielding an extremely large swapping distance as illustrated in Table I. Our method aims to arrange the images along one circular path, therefore minimizing the number of swaps also for dense sets or subsets.

Comparison to SfM. We examined two open-source codes, Bundler [Snavely et al. 2006] and VisualSfM [Wu 2011], to test whether and how well these scenes can be handled by a complete SfM method. Both Bundler and VisualSfM did rather poorly in terms of finding one model with which all which images agree. Table I demonstrates that SfM could almost never recover the latent ring order. VisualSfM's reconstructions are displayed in the Supplementary Material. SIT-ON-FEET and MOTORCYCLE, among the most densely captured sets, were the only ones fully recovered by this software. Bundler failed to fully recover any of the sets.

SfM techniques rely heavily on finding a solid amount of points in precise correspondence between pairs of photos and that a sufficiently large number of such image pairs are available. To better quantify the claim that our RingIt technique relies on weaker priors, we performed two additional experiments. We compared our performance to VisualSfM on our densely captured SIT-ON-FEET set as a function of the number of images in the set and as a function of the image quality. The results are demonstrated in the Supplementary Material.

Scalability and robustness. The almost faultless ordering of our 18 challenging dynamic datasets is strong evidence of our method's robustness. Figure 8(a) illustrates the VisualSFM reconstruction of 11 (out of the 14) images belonging to our BOTTOM-UP set. As the figure demonstrates, our algorithm was examined under noisy conditions where the input images do not reside accurately along a ring. Nevertheless, to quantify the robustness of our spectral approach, we performed Monte Carlo experiments on N input camera locations distributed randomly along the unit circle in $\mathcal{R}^2$. We examined how well the ring ordering performs under an increasing level of noise, or as the input sways away from the assumed circular geometry. Just as our image-based or contour-based distances only approximate the radial, or ring, distances, we consider the Euclidean pairwise distances in order to form a sparse weighted Laplacian. The noise n_t is added to shift the 2D locations along a vector normal to the unit circle. Figure 8(b) displays the swapping distance obtained by running 10000 test cases on increasing values of r_t, where $0 < r_t < 1$. In each test the noise is uniform, that is, $n_t \sim U[-r_t, r_t]$. Figure 8(c) provides an example of one test case. As the figure demonstrates, the spectral method is robust and can recover the correct order almost faultlessly up to a rather large amount of noise.

We also experimented with how well our technique performs "in the wild", that is, with images downloaded from the Internet. We tried our image-based approach with 25 images capturing Obama's inauguration from multiple viewpoints and our contour-based approach with 15 images of different people performing a Tai Chi kick. The results are demonstrated in Figure 9. Even though we

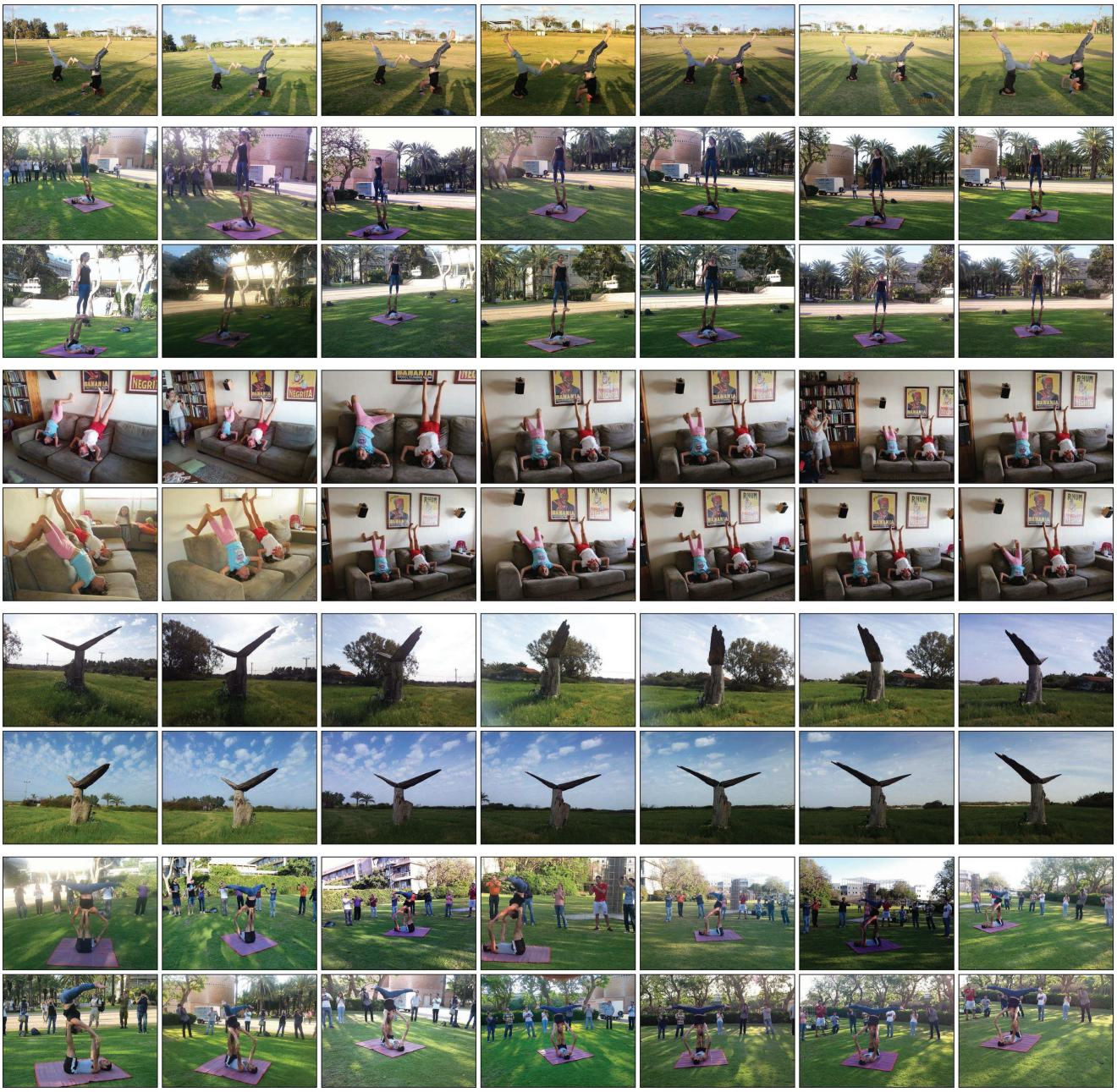

Fig. 10. Image-based RingIt: Partial results of five sets. The output ring order is demonstrated clockwise. Dataset names from top to bottom: HEADSTAND, STAND-ON-FEET, BOTTOM-UP, BIKE&STATUE, BACK BALANCE.

have no exact ground truth with which to compare, the results are clearly reasonable.

Regarding the scalability in N, our algorithm is composed of two independent computations: constructing the dissimilarity matrix and the spectral analysis. Extracting the second and third eigenvectors of a sparse matrix can be computed efficiently for large N, and therefore the dissimilarity measure calculation is the computational bottleneck. The fast and coarse all-pairs measure is quadratic in N, and the more elaborate measure is linear in N. In practice, the running time is dominated by that of NRDC. In our implementation, the NRDC computation of each image pair takes about ten seconds. In Figure 10 we show image-based RingIt. In Figure 11, we show counter based.

8. DISCUSSION AND CONCLUSIONS

We have presented a technique to order an unorganized set of casual photos along a general ring where all cameras look at and see the subject. Although we named it RingIt, the technique is applicable an ellipse or any other star-shaped formation. The key is that it only

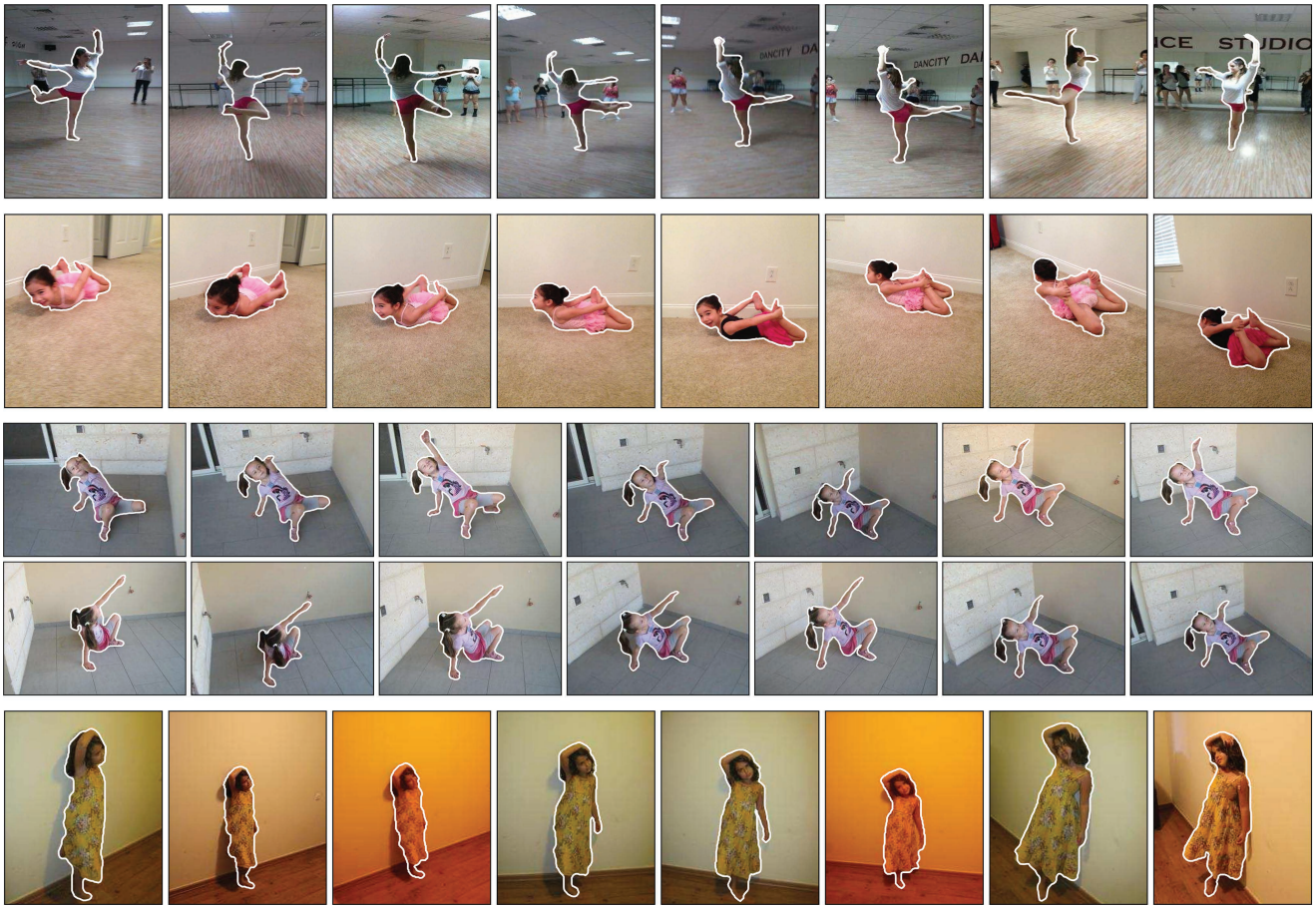

Fig. 11. Contour-based RingIt. Partial results of four sets. Dataset names from top to bottom: ARABESQUE, BACKBEND, BRIDGE, FLAMINGO. Matching silhouettes allows for numerous opportunities, as we are no longer compelled to instantaneous events. For example, BACKBEND represents two different events of the same pose.

requires an analysis of a sparse dissimilarity matrix. Regarding limitations, note that our method relies on having close-enough images in the sense that the nearest neighbors should be sufficiently close to yield a reliable dissimilarity value. This also implies that nearby views are expected to have low dissimilarity values. While this is typically the case, it is not guaranteed due to illumination, occlusions, or other distractors.

We have shown that our method can cope with challenging dynamic settings where a full SfM reconstruction is far from guaranteed. The efficiency of the method is attributed to the fact that we solve a specific problem of ring embedding rather than a generic one aiming at reconstructing the scene. Similarly, we have also shown our ring embedding more effective than the general classical MDS embedding which uses a full dissimilarity matrix. We conclude that having prior knowledge about the expected geometry of the embedding helps to make the process more robust and efficient.

Encouraged by our results, we plan to study more complex geometrical shapes of the expected embedding. We also intend to incorporate morphing and segmentation techniques to create appealing bullet-time displays. We would additionally like to investigate the creation of paths that do not necessarily traverse the entire input images, but only a subset of certain properties. Furthermore, we would like to extend our approach to images that are co-located in space and time, where the dynamic event could be generalized to express a chronology of momentary occurrences of interest, or *time slices*. The desired mode of exploration would consequently be to travel along multiple rings centered about the event, where each ring corresponds to one time slice. The spectral approach could potentially extend beyond 2D to higher dimensions to allow separation in both space and time.

ELECTRONIC APPENDIX

The electronic appendix to this article can be accessed in the ACM Digital Library.

REFERENCES

I. Arev, H. S. Park, Y. Sheikh, J. K. Hodgins, and A. Shamir. 2014. Automatic editing of footage from multiple social cameras. *ACM Trans. Graph.* 33, 4.

A. Arpa, L. Ballan, R. Sukthankar, G. Taubin, M. Pollefeys, and R. Raskar. 2013. Crowdcam: Instantaneous navigation of crowd images using angled graph. In *Proceedings of the International Conference on 3D Vision (3DV'13)*. 422–429.

L. Ballan, G. J. Brostow, J. Puwein, and M. Pollefeys. 2010. Unstructured video-based rendering: Interactive exploration of casually captured videos. *ACM Trans. Graph.* 29, 4.

T. Basha, Y. Moses, and S. Avidan. 2012. Photo sequencing. In *Proceedings of the European Conference on Computer Vision (ECCV'12).* 654–667.

D. Batra, A. Kowdle, D. Parikh, J. Luo, and T. Chen. 2010. iCoseg: Interactive co-segmentation with intelligent scribble guidance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'10).* 3169–3176.

M. Belkin and P. Niyogi. 2003. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Comput.* 15, 6, 1373–1396.

A. A. E. Brouwer and W. H. Haemers. 2012. *Spectra of Graphs*. Springer.

G. Cormode. 2003. Sequence distance embeddings. Ph.D. thesis, University of Warwick. http://webcat.warwick.ac.uk/record=b1663364∼S1.

V. De Silva, D. Morozov, and M. Vejdemo Ohansson. 2011. Persistent cohomology and circular coordinates. *Discr. Comput. Geom.* 45, 4, 737–759.

J.-M. Frahm, P. Fite-Georgel, D. Gallup, T. Johnson, R. Raguram, C. Wu, Y.-H. Jen, E. Dunn, B. Clipp, S. Lazebnik et al. 2010. Building Rome on a cloudless day. In *Proceedings of the 11th European Conference on Computer Vision (ECCV'10).* 368–381.

J.-Y. Guillemaut, J. Kilner, and A. Hilton. 2009. Robust graph-cut scene segmentation and reconstruction for free-viewpoint video of complex dynamic scenes. In *Proceedings of the 12th International Conference on Computer Vision (ICCV'09).* 809–816.

Y. Hacohen, E. Shechtman, D. B. Goldman, and D. Lischinski. 2011. Non-rigid dense correspondence with applications for image enhancement. *ACM Trans. Graph.* 30, 4.

Y. Hacohen, E. Shechtman, D. B. Goldman, and D. Lischinski. 2013. Optimizing color consistency in photo collections. *ACM Trans. Graph.* 32, 4.

K. M. Hall. 1970. An r-dimensional quadratic placement algorithm. *Manag. Sci.* 17, 3.

K. Heath, N. Gelfand, M. Ovsjanikov, M. Aanjaneya, and L. J. Guibas. 2010. Image webs: Computing and exploiting connectivity in image collections. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'10).* 3432–3439.

T. Kanade, P. Rander, and P. Narayanan. 1997. Virtualized reality: Constructing virtual worlds from real scenes. *MultiMedia* 4, 1, 34–47.

I. Kemelmacher Hlizerman, E. Shechtman, R. Garg, and S. M. Seitz. 2011. Exploring photobios. *ACM Trans. Graph.* 30, 4.

A. Kushal, B. Self, Y. Furukawa, D. Gallup, C. Hernandez, B. Curless, and S. M. Seitz. 2012. Photo tours. In *Proceedings of the International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT'12).*

J. A. Lee and M. Verleysen. 2005. Nonlinear dimensionality reduction of data manifolds with essential loops. *Neurocomput.* 67, 29–53.

H. Ling and D. W. Jacobs. 2007. Shape classification using the inner-distance. *IEEE Trans. Pattern Anal. Mach. Intell.* 29, 2, 286–299.

X. Lu, C. Wang, J.-M. Yang, Y. Pang, and L. Zhang. 2010. Photo2trip: Generating travel routes from geo-tagged photos for trip planning. In *Proceedings of the International Conference on Multimedia (MM'10).* 143–152.

A. Oliva and A. Torralba. 2006. Building the gist of a scene: The role of global image features in recognition. *Progress Brain Res.* 155, 23–36.

R. Pless and I. Simon. 2001. Embedding images in non-flat spaces. In *Proceedings of the International Conference on Imaging Science, Systems, and Technology (CISST'01).*

C. Rother, V. Kolmogorov, and A. Blake. 2004. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.* 23, 3, 309–314.

F. Schaffalitzky and A. Zisserman. 2002. Multi-view matching for unordered image sets, or "how do I organize my holiday snaps?" In *Proceedings of the 7th European Conference on Computer Vision (ECCV'02).* 414–431.

N. Snavely, R. Garg, S. M. Seitz, and R. Szeliski. 2008. Finding paths through the world's photos. *ACM Trans. Graph.* 27, 3.

N. Snavely, S. M. Seitz, and R. Szeliski. 2006. Photo tourism: Exploring photo collections in 3D. *ACM Trans. Graph.* 25, 3, 835–846.

Visual Geometry Group. 2004. Multiview and Oxford Colleges building reconstruction. http://www.robots.ox.ac.uk/%7Evgg/data/data-mview.html.

G. Wan, N. Snavely, R. D. Cohen, Q. Zheng, B. Chen, and S. Li. 2012. Sorting unorganized photo sets for urban reconstruction. *Graph. Models* 74, 1, 14–28.

C. Wu. 2011. VisualSFM: A visual structure from motion system. http://ccwu.me/vsfm/.

K. Yucer, A. Jacobson, A. Hornung, and O. Sorkine. 2012. Transfusive image manipulation. *ACM Trans. Graph.* 31, 6, 176.

C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski. 2004. High-quality video view interpolation using a layered representation. *ACM Trans. Graph.* 23, 3, 600–608.

Received July 2014; revised December 2014; accepted February 2015